



MINISTERSTVO ŠKOLSTVÍ,  
MLÁDEŽE A TĚLOVÝCHOVY



OP Vzdělávání  
pro konkurenceschopnost



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

# TECHNICKÁ UNIVERZITA V LIBERCI

Fakulta mechatroniky, informatiky a mezioborových studií



## Generátor signálu snímače otáček motoru

Bakalářská práce

Lubomír Vích

Liberec

2011

Tento materiál vznikl v rámci projektu ESF (CZ.1.07/2.2.00/07.0247)

**Reflexe požadavků průmyslu na výuku v oblasti automatického řízení a měření,**  
KTERÝ JE SPOLUFINANCOVÁN EVROPSKÝM SOCIÁLNÍM FONDEM A STÁTNÍM ROZPOČTEM ČESKÉ REPUBLIKY.



# TECHNICKÁ UNIVERZITA V LIBERCI

## Fakulta mechatroniky, informatiky a mezioborových studií

Studijní program: B 2612 – Elektrotechnika a informatika

Obor: 2612R011 – Elektronické informační a řídicí systémy

# Generátor signálu snímače otáček motoru

## Engine-speed Sensor Signal Generator

### Bakalářská práce

Autor: **Lubomír Vích**

Vedoucí práce: Ing. Jan Koprnický, Ph.D.

Konzultant: –

V Liberci 18. května 2011



## Prohlášení

Byl(a) jsem seznámen(a) s tím, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu TUL.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Bakalářskou práci jsem vypracoval(a) samostatně s použitím uvedené literatury a na základě konzultací s vedoucím bakalářské práce a konzultantem.

Datum

Podpis



## Poděkování

Tímto bych chtěl poděkovat vedoucímu práce Ing. Janu Koprnickému, Ph.D., za spolupráci na tvorbě této bakalářské práce. Cením si jeho rychlého a aktivního přístupu při jakémkoli dotazu a děkuji mu za propůjčené přístroje i za jeho konzultace.



## Abstrakt

Tato bakalářská práce se zabývá návrhem generátoru signálu pro snímač otáček automobilu Škoda Superb 1,9 TDI první generace. Práce obsahuje návrh generátoru a jeho realizaci ve dvou variantách. První je zapojení s obvodem 555 a druhá je naprogramovaný procesor v C++. Zabývá se zapojením jednotlivých součástí v návrhu a popisuje jejich funkci. Seznamuje s výrobou a návrhem generátoru v různých programech. V druhé části popisuje výrobu programovatelných procesorů a jejich programování. Vybraný procesor nese název ATtiny45 od firmy AVR. Také se zabývá zapojením jednotlivých snímačů a seznamuje s důležitými prvky automobilu pro realizaci této práce.

*Klíčová slova: snímače otáček motoru, řídicí jednotka automobilu, programátor AXIS PRESTO, AVR procesory*

## Abstract

This thesis describes the design of the signal generator for revolution sensor of Skoda Superb 1.9 TDI the first generation. The work includes design of the generator and its realization in two variants. The first is connection with 555 circuit and second is processor programmed in C++. It deals with the involvement of individual components in the design and describes their function. It acquaints with the manufacturing and design of the generator in various programs. In the second part it describes manufacturing of programmable processors and their programming. We have selected the ATtiny45 microcontroller manufactured by AVR. It also examines the connection of individual sensors and introduces important elements of the vehicle for the realization of this work.

*Keywords: engine speed sensor, car control unit, AXIS PRESTO programmer, AVR CPU*



# Obsah

<b>Prohlášení</b>	<b>4</b>
<b>Poděkování</b>	<b>5</b>
<b>Abstrakt</b>	<b>6</b>
<b>Abstract</b>	<b>6</b>
<b>Obsah</b>	<b>8</b>
<b>1 Úvod</b>	<b>10</b>
<b>2 Automobil a elektronika</b>	<b>11</b>
2.1 Škoda Superb . . . . .	11
2.2 Řídicí jednotka . . . . .	11
2.3 Sběrnice v automobilech . . . . .	12
2.3.1 Historie sběrnice CAN-BUS . . . . .	13
2.3.2 Přenos dat v CAN . . . . .	13
2.3.3 Obvodové řešení CAN-BUS . . . . .	14
2.3.4 Sběrnice LIN . . . . .	14
<b>3 Snímače otáček motoru</b>	<b>15</b>
3.1 Funkce snímačů . . . . .	15
3.2 Typy snímačů . . . . .	15
3.2.1 Induktivní snímač . . . . .	16
3.2.2 Hallův snímač . . . . .	16
<b>4 Návrh generátoru pomocí obvodu 555</b>	<b>16</b>
4.1 Obvod 555 . . . . .	16
4.1.1 Monostabilní klopný obvod . . . . .	17
4.1.2 Astabilní klopný obvod . . . . .	18
4.2 Praktická část . . . . .	18
4.2.1 Test otáčkoměru . . . . .	18



4.2.2	Obvodová realizace generátoru . . . . .	19
4.3	Prostředí VISA . . . . .	21
4.3.1	Základní informace . . . . .	21
4.3.2	Zdroje VISA . . . . .	22
4.3.3	Programování VISA . . . . .	23
4.3.4	Základní princip programu . . . . .	24
<b>5</b>	<b>Návrh generátoru pomocí procesoru</b>	<b>24</b>
5.1	Atmel a mikročipy AVR . . . . .	24
5.2	Procesor ATtiny45 . . . . .	25
5.2.1	Popis ATtiny 45 . . . . .	25
5.2.2	Pomocný obvod 4053D . . . . .	26
5.2.3	Zapojení procesoru . . . . .	26
5.3	Programování procesoru . . . . .	28
5.3.1	Programátor AXIS PRESTO . . . . .	28
5.3.2	Programovací jazyk C++ . . . . .	30
5.3.3	Programování . . . . .	31
	<b>Závěr</b>	<b>34</b>
	<b>Literatura</b>	<b>35</b>
	<b>Přílohy</b>	<b>37</b>
<b>A</b>	<b>Návrh generátoru pomocí obvodu 555</b>	<b>37</b>
A.1	Seznam součástek (řešení s 555) . . . . .	38
<b>B</b>	<b>Návrh generátoru pomocí procesoru ATtiny45</b>	<b>39</b>
B.1	Seznam součástek (řešení s ATtiny45) . . . . .	40
B.2	Návrh plošného spoje pro generátor s procesorem ATtiny45 . . . . .	41
B.3	Vyrobená deska generátoru s procesorem ATtiny45 . . . . .	42
B.4	Textový výpis z programu AVR studio . . . . .	43
<b>C</b>	<b>Obsah kompaktního disku (CD)</b>	<b>46</b>



## Seznam obrázků

1	Propojení vodičů a sběrnic . . . . .	13
2	Snímání zubů a umístění snímače na kotouči rotoru. [9] . . . . .	16
3	555 v zapojení MKO a v pravo AKO [4] . . . . .	18
4	Vykreslený signál v programu Intuilink Waveform Editor . . . . .	20
5	CMOS 4017 zapojení a jednotlivé výstupní piny. [6] . . . . .	21
6	Propojení zařízení přes GPIB. [5] . . . . .	23
7	Procesor ATtiny 45 [8] . . . . .	25
8	Rozmístění programovacích pinů plošný spoj a programátor. [10] . . . . .	28
9	Staré a nové provedení programátoru. [12] . . . . .	29

## Seznam tabulek

1	Přehled benzínových a dieslových variant motorů pro Škodu Superb. [1] . . .	12
2	Popis pinů pro obvod 555 . . . . .	17
3	Seznam součástek (řešení s 555) . . . . .	38
4	Seznam součástek (řešení s ATtiny45) . . . . .	40





# 1 Úvod

Tato práce se zabývá elektronikou automobilu, programováním procesoru a celkovou výrobou generátoru. Pracuje se na emulátoru Škody Superb, kde je záměrem zprovoznit otáčkoměr na palubní desce. Cílem této práce je zprovoznit další část emulátoru, aby nebránila v rozvoji a zprovozňování zbylých komponentů na panelu. Nejdříve je popsána funkce základních elektronických komponentů jako jsou řídicí jednotka nebo sběrnice v automobilech. Seznamuje se snímači otáček a nakonec s celkovou výrobou generátoru a uvedením do provozu pomocí diagnostických přístrojů. Generátor bude navržen ve dvou provedeních a obě budou uplatněny na otáčkoměru panelu elektroinstalace Škody Superb. První bude pomocí časovače a samotných čítačů a druhé bude na základě programování procesoru. Obě tato řešení budou vyzkoušena a v závěru zhodnoceny celkové dosažené výsledky a jejich případné rozšíření.



## 2 Automobil a elektronika

### 2.1 Škoda Superb

Škoda Auto [1] v roce 2001 uvedla na trh automobil s názvem Škoda Superb, který má zastávat pozici velkého rodinného vozu. Navazuje na tradici automobilů Superb, které automobilka vyráběla před druhou světovou válkou jako luxusní modely. Superb je založen na platformě Volkswagenu Passat, ale s jinými rozměry, a vyrábí se v několika variantách. V současnosti je Superb nejluxusnější model u Škoda Auto, navíc s vysokým výkonem.

Jako první z moderních SuperB se na trhu objevilo Superb. Jedná se o 4dvěřovou variantu ve formě sedanu a hatchbacku. Je k dispozici jak v benzínovém provedení tak s dieslovým spalovacím motorem. Základní model je čtyřválcový 1,9 litrů v TDI s výkonem 74 kW, nebo 2 litrový benzínový s 85 kW. Převodovky u Škody Superb B5, typu U3 jsou pětistupňové manuální a šestistupňové manuální. Na trhu jsou také varianty s pětistupňovou automatickou převodovkou. B5 má pouze přední hnací nápravu.

Novější model nese název Škoda Superb B6, Typ 3T. Na trh se dostal v roce 2008. Rozdíl oproti jeho předchůdci je v hnací nápravě, která je u tohoto modelu pouze přední, nebo s pohonem na všechna čtyři kola. Typ převodovky zůstal stejný a vyrábí se také v dieslovém i benzínovém provedení (viz tab. 1/s. 12). Od roku 2009 se vyrábí i ve formě 5dvěřového combi.

V průběhu let bylo auto vylepšováno různými komponenty navíc. Začlenily se nové masky, chladiče, světlomety, dřevěný interiér pro řadu Elegance a mnoho dalšího.

V roce 2010 bylo Superb combi v Británii vyhlášeno nejlepším rodinným vozem na světě.

### 2.2 Řídicí jednotka

Snímače otáček by nemohly fungovat bez řídicí jednotky (dále jen ŘJ). ŘJ pracuje jako paměťový řídicí systém, do kterého si lze naprogramovat prakticky cokoli, co si zákazník u svého vozu přeje. Ovládá například téměř celou autodiagnostiku a zajišťuje funkčnost mnoha systémů, od ABS po automatickou převodovku. V tomto konkrétním případě řídí spalování v motoru v závislosti na otáčkách motoru. ŘJ pracují v elektrickém zapojení s různými zapalovacími soupravami.



Tabulka 1: Přehled benzínových a dieslových variant motorů pro Škodu Superb. [1]

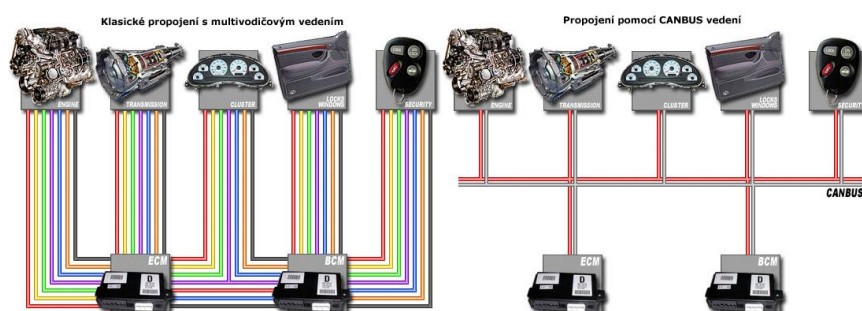
Typ motoru	Popis
1,4 TSI/92 kW	zážehový, přeplňovaný turbodmychadlem, přímé vstřikování paliva
1,8 TSI/118 kW	zážehový, přeplňovaný turbodmychadlem, přímé vstřikování paliva
2,0 TSI/147 kW	zážehový, přeplňovaný turbodmychadlem, přímé vstřikování paliva
3,6 FSI V6/191 kW	zážehový, vidlicový, přímé vstřikování paliva
1,6 TDI CR DPF/77 kW	vznětový, přeplňovaný turbodmychadlem s nastavitelnou geometrií lopatek, přímé vysokotlaké vstřikování paliva common-rail
2,0 TDI CR DPF/103 kW	vznětový, přeplňovaný turbodmychadlem s nastavitelnou geometrií lopatek, přímé vysokotlaké vstřikování paliva common-rail
2,0 TDI CR DPF/125 kW	vznětový, přeplňovaný turbodmychadlem s nastavitelnou geometrií lopatek, přímé vysokotlaké vstřikování paliva common-rail

## 2.3 Sběrnice v automobilech

Sběrnice je síť kabelů, která má zajistit přenos dat nebo řídicích povelů mezi dvěma a více elektronickými zařízeními. Sběrnice mohou být paralelní, jako jsou adresové, datové, řídicí a nebo sériové pro sdílení nebo řízení na vodičích.

Kdyby neexistovaly sběrnice, bylo by celé vozidlo propojeno rozsáhlou sítí kabelů a vodičů (viz obr. 1/s. 13)[2]. Toto řešení není nemožné, ale díky sběrnici, která zajišťuje stejnou funkčnost, ušetříme více místa a získáme jednodušší přístup ke komponentám a případným opravám v elektronice.

Sběrnice už jsou ve výrobě poměrně dlouho. Technologie a elektronické komponenty se stále zdokonalují, jsou rozšířeny po celém světě a sběrnice k tomuto nárůstu určitě pomohly. Bez nich by byly komponenty mnohem složitější. V automobilech je už také mnoho elektroniky, jejíž zapojení by bez sběrnice bylo dost komplikované. Moderní automobil má automatické stahování oken, vyhřívaná zrcátka a mnoho dalšího. Propojení jednotlivými vodiči by bylo určitě finančně náročnější. Sběrnice mimo jiné dovoluje připojit nové zařízení na



Obrázek 1: Propojení vodičů a sběrnic

sebe samotnou. U klasického vedení vodičů by bylo nutné ho připojit sítí kabelů ke každému dalšímu zařízení, které se už v systému nachází.

Sběrnice je pro tyto důvody velmi využívána. V automobilovém průmyslu jsou používány sběrnice typu CAN, LIN, SEP, ACP, atd.

### 2.3.1 Historie sběrnice CAN-BUS

CAN<sup>1</sup> [2] sběrnice je sériová a byla vyvinuta firmou Bosch v roce 1983. Sběrnice sama o sobě je symetrický nebo asymetrický dvou vodičový obvod, který může být odstíněný či neodstíněný. Tato sběrnice byla vyrobena z důvodu zvyšujících se nároků na rychlost a množství přenášených dat. I když tato sběrnice byla navržena pro automobilový průmysl, díky jejím vlastnostem byla brzy po vyrobení použita i v průmyslových aplikacích pro komunikace čidel, nebo dalších řídicích systémů. Sběrnice CAN byla v automobilech poprvé použita v roce 1991, a to ve vozech Mercedes Benz třídy S. Ve Škodě Auto až o 5 let později.

Sběrnice CAN je využívána pro komunikaci senzorů a funkčních jednotek v automobilech. Je označována v průmyslových komunikačních sítích jako fieldbus. Maximální rychlost přenosu dat je 1 Mbit/s.

### 2.3.2 Přenos dat v CAN

V přenosu dat u CAN není třeba centrálního uzlu. Tento systém se nazývá Multi-Master. CAN pracuje tak, že síťový protokol detekuje a opravuje přenosové chyby vzniklé od okolních elektromagnetických polí. CAN odesílá data po rámcích. Každý rámec může obsahovat až 8

<sup>1</sup>Controller Area Network



datových bajtů. Také obsahuje identifikátor, protože u CAN neexistuje žádná adresa, vše se zapisuje do jeho paměti. Zapisují se tam konkrétní zprávy a také důležitost zprávy. Čím nižší je hodnota identifikátoru, tím vyšší je priorita zprávy a má přednost při odesílání na sběrnici. Identifikátor může přijít i do několika zařízení zároveň (mnohonásobný příjem). Pokud přijde zpráva s vyšší prioritou, identifikátor ji vyřídí nejdříve a až potom se vrátí k předchozím. Každý identifikátor je jednoznačný. Tím že v CAN neexistují adresy, je identifikátor jediné možné spojení a poté záleží na konkrétních komponentech, ke kterým zpráva dojde, zda je to pro ně důležité. Rychlosti, se kterými CAN pracuje a jejich možné vzdálenosti, jsou následující:

- Motorová CAN 1 Mbit/s 40 m,
- Multimediální CAN 500 Mbit/s 300 m,
- Komfortní CAN 125 Mbit/s 600 m.

### 2.3.3 Obvodové řešení CAN-BUS

Ke správnému fungování sběrnice CAN je třeba mít připojeno do řídicí jednotky několik obvodů. Nejdůležitější obvod je mikroprocesor. Mikroprocesor dává pokyny pro vysílání zpráv, ale také i zpracovává přijatá data. Dalším obvodem je radič. Jedná se vlastně o filtr, který je napojen na datovou linku protokolu, kde například filtruje zprávy nebo kontroluje rámce. Poslední důležitou částí je budič. I ten má v obvodovém řešení CAN využití. Budič převádí signál z radiče CAN do fyzické vrstvy CAN (napětí, diferenciální, sběrnice).

### 2.3.4 Sběrnice LIN

LIN<sup>2</sup> je sériová sběrnice, navržená pro použití v automobilové technice. Oproti sběrnici CAN je levnější, například také proto, že je zatím méně komerčně prosazena na trhu. V automobilech se používá například pro klimatizace nebo na změnu polohy zrcátek. Stejně se tato sběrnice používá i v jiných odvětvích průmyslu. Můžeme ji najít v různých spotřebních elektronikách a měřicích technikách. Byla vyrobena v roce 1999 a poprvé se do automobilu

---

<sup>2</sup>Local Interconnect Network



dostala v roce 2001. Komunikace je založena na single-master/multiple-slave jednovodičové sběrnici o napětí 12 V. Má na jednom vodiči obousměrnou komunikaci. Maximální přenosová rychlost je až 20 kbit/s a délka 20 m.

## 3 Snímače otáček motoru

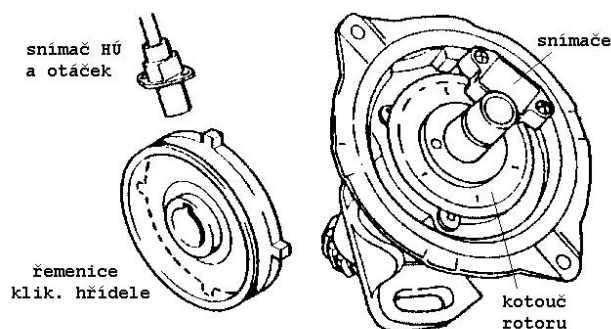
### 3.1 Funkce snímačů

Snímání otáček a poloh v motoru nám zajišťuje přehled o aktuálních otáčkách a polohách hřídelí. Snímače otáček informují o svém stavu řídicí jednotku ve voze. Jsou to jedny z ne důležitějších motorových čidel. V systému řízení motoru se jedná o snímání klikové a vačkové hřídele nebo hřídele vstřikovacího čerpadla, v případě starších systémů také polohy či otáček hřídele rozdělovače. Snímač reaguje na tvar impulzního kola snímače, které je upevněno na hřídeli, a tento signál posílá řídicí jednotce. Kolo má šedesát zubů a z toho jsou dva vynechány. Z těchto signálů dokážeme pomocí výpočtů určit např. úhel zážehu a dobu vstřiku u zážehových motorů nebo dobu a počátek vstřiku ve vznětových motorech.

### 3.2 Typy snímačů

Typů snímačů otáček motoru je několik. Mezi hlavní dva patří induktivní snímač a Hallův snímač. Na našem přípravku Škody Superb 1.9 TDI je použit první zmiňovaný, a to induktivní snímač. Všeobecné výhody těchto snímačů jsou následující:

- vysoká přesnost a stabilita elektrických parametrů,
- kompaktnost (integrovaný nosič kabelů),
- vynikající odolnost proti prachu, oleji, olejové mlze, vibracím a otřesům,
- bezdotykové snímání,
- spolehlivost v náročných provozních podmínkách,
- nízká cena,
- jednoduchá montáž.



Obrázek 2: Snímání zubů a umístění snímače na kotouči rotoru. [9]

### 3.2.1 Induktivní snímač

Tyto typy snímačů pracují s rychlou změnou intenzity magnetického toku okolo jádra, čímž se indukuje elektrické napětí ve vinutí cívky. Rychlé střídání feromagnetických zubů a mezer indukuje při dostatečné rychlosti (např. otáčení impulzního kola ve vinutí cívky) střídavé napětí. Tyto snímače bývají umístěny v bezprostřední blízkosti impulzního kola vždy na pevné části (motoru, převodovky, atd.). Velikost indukovaného napětí je přímou úměrou závislá na velikosti změny magnetického pole závislé na vzdálenosti snímače (např. rychlosti otáčení kola). Mají analogový signál ve tvaru sinusové křivky.

### 3.2.2 Hallův snímač

Tento typ snímače musí být napájen. Napětí se přivádí z ŘJ (5 V, nebo i 12 V) na kostru. Má pravoúhlý signál. Tyto snímače jsou oproti induktivním o něco přesnější. Jsou nezávislé na rychlosti otáčení impulzního kola a mají jemnější rozpoznání polohy.

## 4 Návrh generátoru pomocí obvodu 555

### 4.1 Obvod 555

První varianta navrhnutí generátoru s časovačem je založena na obvodu 555. Integrovaný obvod 555 [4] je jeden z nejlépe konstruovaných integrovaných obvodů se širokým uplatněním. Tento obvod je i po třiceti letech od své výroby stále populární. Vyrábí se v plastovém



pouzdrě DIP8<sup>3</sup> a milionové výrobní série zajistily nízkou cenu na trhu. Nejčastější zapojení 555 je jako časovač.

Časovač 555 se vyrábí ve dvou provedeních. První klasické je CMOS a druhé je známé pod označením 556. 555 pracuje v rozsahu 4–18 V, přičemž nevyžaduje souměrné napětí jako operační zesilovač. Verze CMOS si vystačí i s nižším napětím. Na výstupu 555, který je vyveden z pinu 3, je k použití i velký proud až 200 mA. Piny 1 a 8 slouží k napájení. Na pin 1 se připojuje záporný pól a na pin 8 kladný pól napájení. Pro napájení stačí nesymetrický zdroj. K blokování funkce je u obvodu 555 pin 4. Napěťový dělič rozděluje napětí na třetiny a na vývodu 5 se nachází 2/3 napájecího napětí. Pokud nechceme měnit poměr napětí, je třeba zvlášť u CMOS verze připojit 10 nF kondenzátor. Podrobnější funkce jednotlivých pinů je zapsána v tabulce (viz tab. 2/s. 17).

Dvě nejčastější zapojení 555 jsou monostabilní klopný obvod a astabilní klopný obvod.

Tabulka 2: Popis pinů pro obvod 555

Pin	Označení	Popis
1	GND	Uzemnění (ground) obvodu, 0 V.
2	TRIG	Spouštění (trigger), vstup druhého (zapínacího) komparátoru.
3	OUT	Výstup (output) obvodu.
4	RESET	Nulovací (reset) vstup, umožňuje nulování KO nezávisle na vstupech.
5	CTRL	Řídící (Control) napětí. Ovlivňuje překlápění komparátorů.
6	THR	Práh (threshold), vstup prvního (vypínacího) komparátoru.
7	DIS	Vybíjení (discharge), kolektor vybíjecího tranzistoru.
8	VCC	Kladné, napájecí napětí v rozsahu 4,5 V až 15 V.

#### 4.1.1 Monostabilní klopný obvod

Označuje se jako MKO a má jeden stabilní stav, ze kterého je možné jej přepnout do stavu nestabilního. Obvod se sám po určité době přepne zpět do stabilního stavu. Tento typ obvodu je možné použít například jako zpožďovací prvek.

K tomuto zapojení 555 (viz obr. 3/s. 18)[4] postačí dvě vnější součástky, nepočítá-li se

<sup>3</sup>Pouzdro pro integrované obvody s osmi vstupy



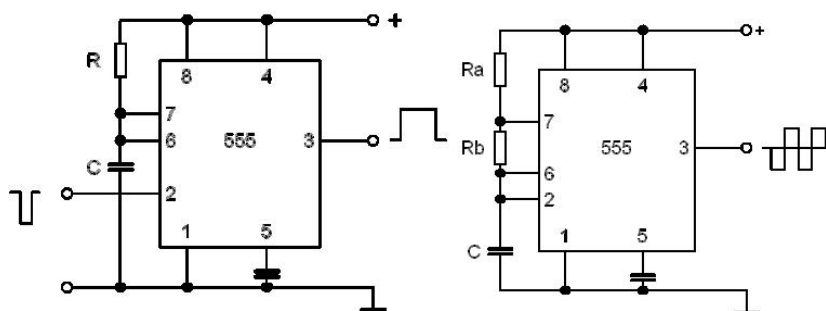


blokovací kondenzátor, který je třeba vždy zapojit. Jedná se o jednodušší zapojení tohoto multifunkčního obvodu.

Napájení se přivádí k pinům 8 a 1. Za výstup se považuje pin 3. Kondenzátor ošetřuje invertující vstup komparátoru na pinu 5. Obvyklá hodnota kondenzátoru se pohybuje v rozmezí 10 až 100 nF.

#### 4.1.2 Astabilní klopný obvod

Značí se jako AKO a nemá žádný stabilní stav, takže tyto obvody neustále oscilují (překlápí se) mezi prvním a druhým stavem. Využívají se jako impulzní generátory nebo blikáče. AKO se dá realizovat pomocí diskretních součástek za použití dvou tranzistorů. Zapojení AKO (viz obr. 3/s. 18)[4] je oproti předchozímu trochu odlišné. Ve schématu nám přibude jeden rezistor. Kondenzátor zůstal na pinu 6, ale kladné napětí se přivádí přes dva rezistory. Kondenzátor je na počátku vybitý a začne se nabíjet po přivedení 2/3 napájecího napětí.



Obrázek 3: 555 v zapojení MKO a v pravo AKO [4]

## 4.2 Praktická část

### 4.2.1 Test otáčkoměru

Na přípravku se žádná ozubená kola nenachází, a proto bylo třeba zvolit jinou cestu. Varianta byla nahradit snímač generátorem, na kterém se nastavil obdélníkový průběh šedesáti pulzů a dva z nich se vynechaly. Toto by mělo nahradit ozubené kolo ve skutečném automobilu. Generátor se připojil na osciloskop, kde lze pozorovat průběh. Generátory byly původně na výběr dva: Hewlett-Packard 33120A nebo Agilent 33120A. Nakonec byl vybrán a použit druhý zmíněný.



Nejprve bylo třeba nastavit generátor na požadovaný průběh. Na samotném generátoru nastavit signál o šedesáti pulzech se dvěma vynechanými, dodržet střidu 71 % a amplitudu 5 V (tyto parametry vychází z katalogové hodnoty Škoda Auto [1]). Bylo potřeba na generátoru nastavit požadovaný průběh. Pro ladění/nastavení je možno použít funkce Burst, která dokáže signály přerušovat. Udržet ji při stejné periodě, aniž by se změnila frekvence nebo střída, je ale hodně složité.

Ve zprávě [3] pan Vít píše, že se tento signál dá vymodelovat přes program Intuilink Waveform Editor. Díky jednoduchosti tohoto programu se tento průběh dal rychle vytvořit. Bylo potřeba obdélníkového signálu a v daném poměru 58/60 se nasimuloval za offsetu 1 průběh 58 pulzů. Zbylé dva se nahradily stejnosměrnou složkou a offsetem 0 ve zbylém poměru 2/60. Výsledný průběh vypadal následovně (viz obr. 4/s. 20).

Tento průběh je třeba nahrát z počítače do generátoru a poté vyslat do řídicí jednotky. Pomocí sériového portu RS-232<sup>4</sup> lze nahrát tento signál do paměti generátoru. Při nahrávání se ovšem objevilo několik problémů s kompatibilitou, a proto se zvolila varianta s GPIB<sup>5</sup>. I poté se nedařilo signál přenést. Generátor na počítači nebyl vidět. Díky programu Agilent VISA byl tento problém vyřešen. Hotový průběh se nahrál do paměti generátoru, který měl všechny parametry v sobě přesně nastavené podle programu. Po propojení s osciloskopem byl viděn požadovaný průběh. Generátor byl zapojen na kabely od snímače a ty začaly vysílat signál. Otáčkoměr reagoval, ovšem vydržel pracovat maximálně 2 sekundy. ŘJ automobilu pozná, že auto ve skutečnosti nejede a otáčkoměr vyřadí z provozu. Pravděpodobně nemá ŘJ dostatek informací, aby model automobilu mohl ve skutečnosti jet. Tento problém mohou také ovlivňovat špatně nastavené parametry, jako je teplota chladicí kapaliny, teplota nasávaného vzduchu atd.

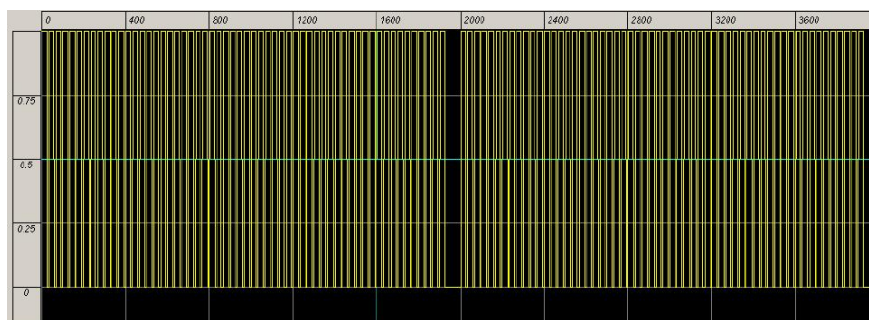
#### 4.2.2 Obvodová realizace generátoru

Obvod s 555 je navrhnut ze tří částí. V první části je stabilizátor, který stabilizuje z 12 V na 5 V, ve druhé je generátor s obvodem 555 a třetí impulzní část s dvěma CMOS čítačema.

Na začátku zapojení je klasické napětí 12 V, které vychází z autobaterie vozidla. To je třeba snížit na 5 V, aby s ním mohla pracovat TTL logika a další obvody ve schématu. Toho

<sup>4</sup>Recommended Standard 232 - sériový port

<sup>5</sup>General Purpose Interface Bus



Obrázek 4: Vykreslený signál v programu Intuilink Waveform Editor

docílíme stabilizátorem na 5 V typu 78L05 se dvěma odrušovacími kondenzátory. Do schématu jsou také zapojeny usměrňovací diody. Výsledný signál nyní vstupuje do obvodu 555, který je zapojen jako časovač v astabilním režimu. Zapojuje se (viz obr. 3/s. 18)[4] s jedním 10 nF kondenzátorem z pinu 5 na GND. Dále obsahuje dva rezistory a jeden kondenzátor. Poměr rezistorů  $RA$  a  $RB$  určuje střidu, kdy střída je poměr délky pulzu a délky mezery. Pokud je střída 50 % (nebo někde i značená střída 1:1) je stejná délka pulzu a stejná délka mezery. Kondenzátor na pinu 2 a 5 řídí frekvenci posílaných pulzů. Všechny tyto součástky je třeba přesně vypočítat, aby byly splněny parametry, které udává dokumentace. Poměr pulzů vypočítáme z následujících vztahů:

$$T1 = 0,693 \cdot (R1 + R2) \cdot C1 \quad (1)$$

$$T2 = 0,693 \cdot R2 \cdot C1 \quad (2)$$

Další vzorec dopočítává možnou frekvenci, která se nám při každé změně při výpočtu odporu mění.

$$f = 1,44 / (R1 + 2 \cdot R2) \cdot C1 \quad (3)$$

Po správném výpočtu a nastavení obvodu 555 vstupuje signál z pinu 3 do dvou CMOS čítačů typu 4017. První z nich je nastaven jako jednotky a druhý jako desítky. Je-li třeba nastavit 58 pulzů, potom nastavíme na prvním obvodu číslo 8 a na druhém číslo 5. Tímto se vyřeší problém s omezením průchodu dalších pulzů. Je třeba propustit ještě dva pulzy, aby se generátor choval přesně jako reálný snímač otáček. Další dva pulzy se nastaví na čítačích jako průchozí, přičemž Druhý průchozí pulz je též nastaven jako resetovací. Zbylá





softwarově propojit hardwarové produkty s různými softwary jedním systémem, pomocí kterého budou spolu moci komunikovat a spolupracovat.

Kromě propojování různých přístrojů a jejich komunikace lze použít VISA k vývoji zcela nových I/O aplikací a přístrojových ovladačů, které jsou kompatibilní se standardy VXI-plug&play. Takto vyvinuté ovladače mohou realizovat systémové programy na úrovni VISA I/O.

VISA je kompatibilní s HP-UX<sup>6</sup> a se všemi systémy Windows od Windows 95. Ve Windows podporuje rozhraní GPIB, VXI, GPIB-VXI, serial (RS-232) a rozhraní LAN. Podpora rozhraní v HP-UX je stejná kromě serial (RS-232).

### 4.3.2 Zdroje VISA

Za zdroj se považuje jakékoliv zařízení, se kterým může VISA navázat komunikaci (např. napojené po LAN, nebo přes GPIB). VISA definuje šest typů tříd zdrojů, které mohou být implementovány do systému VISA. Třídy se liší způsobem připojení, komunikace, programování ale i způsobem použití.

Každá třída obsahuje:

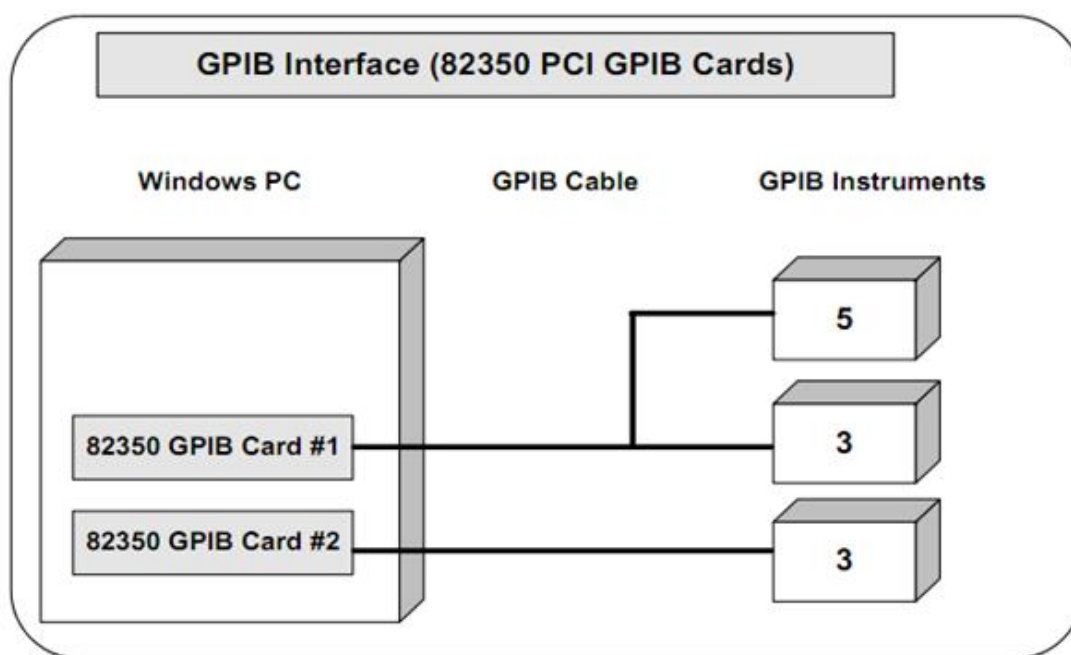
- Attributes (atributy) – determinují stav a relaci zdroje nebo nastavují zdroji či relaci určitý stav.
- Events (události) – slouží ke komunikaci s aplikacemi.
- Operations (operace) – funkce, které mohou být použity pro jednotlivé třídy.

Typy tříd zdrojů:

- Instrument control
- GPIB bus interface
- Memory Access
- VXI mainframe backplane
- Servant device - Side Resource

---

<sup>6</sup>Unix od firmy Hewlett-Packard



Obrázek 6: Propojení zařízení přes GPIB. [5]

- TCPIP Socket

GPIB a VXI jsou hardwarová rozhraní pro měřicí a zkušební přístroje a zařízení, která umožňují přenos dat mezi dvěma nebo více přístroji (viz obr. 6/s. 23). Umožňuje připojení počítače, který může řídit přenos dat. GPIB a VXI velice usnadňují aplikaci VISA komunikaci s přístroji a programování více různých zařízení najednou.

#### 4.3.3 Programování VISA

Agilent VISA lze programovat více programovacími jazyky, mezi ně patří hlavně C/C++, visual Basic a MATLAB. Jako kompatibilní programovací aplikace je možno použít:

- Agilent VEE
- C/C++
- MathWorks MATLAB
- Microsoft Visual Basic 6
- Microsoft Visual Studio.Net (using Visual Basic, Managed C++ or C#)



- National Instruments LABVIEW
- National Instruments LabWindows/CVI

#### 4.3.4 Základní princip programu

K přehledu nad všemi zařízeními a zdroji slouží Resources manager session, je to hlavní relace, která má přehled nad ostatními otevřenými relacemi. Ke komunikaci zařízení či zdrojů na rozhraní slouží Resource session. Relace musí být otevřena pro každý zdroj, se kterým chce uživatel pracovat. Každé zařízení při zapojení získá unikátní jméno, díky kterému je možno ho lehce identifikovat. Výhodou relací je to, že neřeší, kde je jaké zařízení připojeno a díky tomu se přes rozhraní lépe komunikuje a přenáší aplikace.

## 5 Návrh generátoru pomocí procesoru

### 5.1 Atmel a mikročipy AVR

První věcí při programování procesoru je vybrat samotný procesor. V dnešní době je mnoho výrobců různých firem a s různými variantami. K výrobě byl doporučen procesor od firmy Atmel ATtiny45.

Atmel Corporation [7] je firma založená v roce 1984 v USA. Její primární zaměření je polovodičová technika, ale velmi věhlasnou se stala i v oblasti výroby integrovaných obvodů.

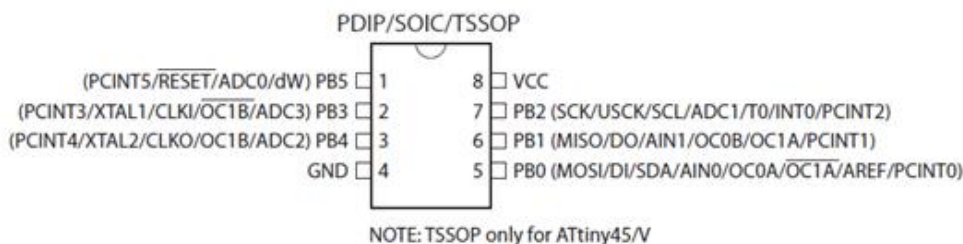
Atmel podniká v mnoha odvětvích přes telekomunikace, průmysl, počítače, zasahuje i do zdravotnictví a vyniká ve výrobě bezpečnostních systémů, kde je známa díky čipovým kartám typu smart card nebo RFID<sup>7</sup>. Firma vlastní několik továren po USA a Evropě s desetitisíci zaměstnanci. Je výrobcem 8bitových mikročipů typu RISC označovaných AVR.

AVR vznikla díky dvěma Norským studentům v roce 1997 a dělí se na tyto základní typy:

- AT90 – nejstarší řada, která už se v dnešní době nevyrábí,
- ATtiny – pro jednoduché a malé obvody. Obsahují až 8 kB flash paměť a vyrábí se v pouzdrech 6-32 pinů,

---

<sup>7</sup>Radio Frequency Identification. Identifikace na rádiové frekvenci.



Obrázek 7: Procesor ATtiny 45 [8]

- ATmega – poslední varianta AVR s větší Flash kapacitou až 256 kB v poudrech 23 – 100 pinů. Široká řada rozhraní.

Zvolili jsme tedy při výběru procesoru firmu Atmel a procesor typu ATtiny. Procesory ATtiny se vyrábějí ve variantách ATtiny 11/12/13/15L/26/28, ATtiny2313 a ATtiny 25/45/85.

Vybrali jsme poslední zmiňovaný ve variantě 45.

## 5.2 Procesor ATtiny45

### 5.2.1 Popis ATtiny 45

Druhé řešení generátoru bylo pomocí programovatelného procesoru. Za vhodný procesor byl zvolen 8bitový mikroprocesor ATtiny45 (viz obr. 7/s. 25). Tento procesor od firmy Atmel zapadá do řady mikrořadičů AVR, které mají 32 registrů a umožňují nám paralelní přístup. Procesory se vyznačují vysokým výkonem a nízkou spotřebou. Obvody AVR obsahují dvě paměti. První z nich je programovatelná flash paměť typu OnChip s kapacitou 4096 bajtů a druhá datová typu EEPROM (elektricky mazatelná programovatelná paměť), u které je možné programovat jejich systém, o velikosti 256 bajtů. Při procesu programování nemusíme připojovat napěťové vodiče. V našem případě se jedná o SMD procesor s šesti univerzálními I/O linkami. Maximální frekvence je 20 MHz, napájecí napětí v rozsahu 2,7 – 5,5 V a teplotní rozsah –40 až 85 °C. Dále obsahuje 4kanálový 10bitový A/D převodník, časovač s vnitřním oscilátorem, software s úsporným režimem a OnChip analogový komparátor.

Obvod se skládá z osmi vývodů. Na osmém je klasicky přivedeno napájecí napětí VCC a na čtvrtém GND se připojuje k zemi. Zbývají piny PB0–PB5. Port B je 6bitový řídicí I/O port s interními rezistory.





Reset neboli nulovací vstup se spustí v případě, že úroveň na pinu bude menší než minimální nastavená délka pulzu. Resetovací pin může být použit také jako slabší I/O pin.

Po zapojení ATtiny45 do obvodu je třeba zachovat i určité minimální parametry. Dokumentace procesoru [8] udává, že mohou nastat problémy s EEPROM pamětí. Problém může nastat při nízké taktovací frekvenci, nebo nízkém napětí. Proto bychom se měli vyvarovat používat paměť EEPROM pod frekvenci 1 MHz a napětí držet nad 2 V.

### 5.2.2 Pomocný obvod 4053D

K zapojení procesoru je také třeba obvod typu CMOS 4053D, který nám plní funkci bypassu nebo výhybky, kvůli omezení spínacích prvků na procesor. Jedná se o analogový multiplexer s čipem, který obsahuje tři analogové spínače. Tento čip se dá zapojit několika způsoby. Pokud se zapojí správně, má malé zkreslení a omezí špatné přepínání. V našem případě plní funkci dvousměrného přepínače a je tam zapojený tak, že přepíná jeden pin na dva různé kontakty. Pokud připojíme programátor, dokáže si přepnout kontakty na samotný programátor a můžeme programovat procesor. Procesor se programuje sériově. Při podržení resetu, se přepne na analog, který přepíná. Jako u všech CMOS je důležité, aby všechny nepoužité piny byly propojeny na zem, nebo nějakou logickou úroveň.

### 5.2.3 Zapojení procesoru

Procesor ATtiny45 (viz obr. 7/s. 25), pracuje v daném napětovém rozsahu. Použitá verze pracuje v rozmezí 2,7 – 5,5 V a druhá verze ATtiny45-D pracuje od 1,8 V. Ve schématu je také použita výhybka CMOS 4017, která ale neumí vygenerovat vyšší napětí, proto je použit 5V stabilizátor typu 78L05 pro napájení ze vstupních 12 V vycházejících z autobaterie. V obvodu působí celkem malý proud do 10 mA, takže by neměl nastat problém s přehřátím stabilizátoru. Před a za stabilizátorem jsou připojeny dva blokovací kondenzátory o 100 nF, které zamezují kolísání napětí. Pokud není přiřazena obvodu daná frekvence, může se stabilizátor dostat do harmonické regulace tím způsobem, že začne kmitat sám od sebe a může se zničit. To nastává, pokud kondenzátory ke stabilizátoru nejsou připojeny. Často se za stabilizátor dává elektrolytický kondenzátor, ale u podobně malých proudů se dá použít keramický. Pokud by v obvodu byla použita například nějaká LED dioda nebo žárovka, je dobré použít elektrolytický kondenzátor. Kondenzátory působí při zapnutí jako zkrat, tím pádem proud



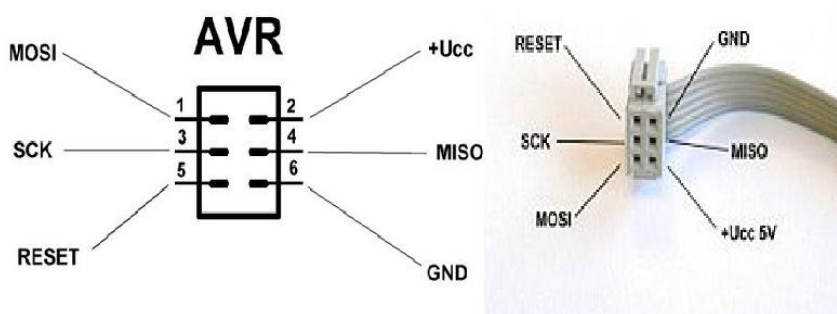
v obvodu nejprve teče a postupně se vyrovná.

Zenerova dioda se zapojuje do obvodu jako ochrana před vznikem přepětí. Zde je použita dioda s rozsahem na 15 V.

U zapojení procesoru se mezi 4. a 8. pinem nachází blokovací kondenzátor, který zamezuje dalšímu kolísání proudu a vedení indukčnosti. Provedení desky je celkem malé, ale u větších desek se při vedení na větší vzdálenosti může indukčnost projevit. Kondenzátor parazitní indukčnost eliminuje a zároveň se chová jako zdroj procesoru. Pokud napětí kolísá, tak kondenzátor malou chvíli napětí udrží a tím působí jako krátkodobý zdroj. Na 1. pinu je negovaný reset. Ten je možné přednastavit, nicméně kvůli programátoru je nastaven na původní hodnotu v resetu. Pokud je v log 1, tak běží, naopak pokud je ve stavu log 0, tak se zastaví. Krystalový oscilátor tam zapojen není. Je totiž obsažen uvnitř v RC oscilátoru a dá se kalibrovat. Nastavit ho lze od 0 do 10 MHz a některé verze zvládnou i 20 MHz. Tato frekvence je už poměrně vysoká, a proto se v tomto zapojení nepoužívá.

Procesor je propojený s výhybkou, která je zde připojena kvůli programátoru. Některé piny jsou vstupní a jiné fungují jako výstupní. Pinům, které jsou vstupní, mohou být přiřazeny odlišné hodnoty a programátor by tam nehrál žádnou roli. Proto je zapojena do obvodu výhybka, která rozšíří vstupy a výstupy. Pokud je připojen do konektoru SV1 napájecí pin, tak v momentě programování se reset sepne na log 0, což způsobí zastavení procesoru, ale zároveň se přepnou všechny vstupy výhybky. Výstupy A, B a C se přepnou z log 1 do log 0. V případě že je reset v log 0, výstupy jsou X0, Y0 a Z0, pokud v log 1 tak X1, Y1 a Z1. Pin INH se chová jako kontrola, která sleduje, zda vyhybka běží, nebo neběží. Takže pokud se připojí programátor a zapne se programování, reset je nastaven v log 0 a tím se jádro procesoru zastaví. V tu chvíli se sepnou piny na výhybce. Jádro procesoru stojí, ale procesor obsahuje řadič SPI, který dokáže přehrát vnitřní flash paměť. SPI řadič je sběrnice, která dokáže komunikovat a zároveň přepisovat paměť. Toto vylepšení, které vynalezla samotná firma Atmel, vede k zjednodušení postupu, takže není třeba používat složitějšího paralelního programování. Takt procesoru SCK musí být maximálně čtvrtina frekvence procesoru, aby to sběrnice stíhala zpracovávat. Analogový vstup a PWM<sup>8</sup> výstup jsou vyvedeny přes odpor. Funkcí odporů v zařízení je zachránit ho v případě, že se zkratují kontakty. U PWM odpor chrání obvod a u analogového vstupu ochraňuje externí zařízení. K analogovému vstupu je

<sup>8</sup>Pulse Width Modulation – Pulzně šířková modulace



Obrázek 8: Rozmístění programovacích pinů plošný spoj a programátor. [10]

připojena zenerova dioda, která má také chránící funkci. Kdyby do obvodu šlo větší napětí, může přes ni obvod zkratovat, čímž se zamezí zničení procesoru.

### 5.3 Programování procesoru

AVR procesor je třeba naprogramovat, a to pouze za pomoci vhodného programátoru. Programátorů mikroprocesorů je celá řada. Programátor se napojí do obvodu a poté se může nahrát příslušný software. Známý a často používaný je program AVR studio vyvinutý firmou Atmel. Vytvářet program v AVR studiu je možné pomocí jazyku C/C++ nebo Assembleru. Po kompilaci se pomocí programátoru nahraje program do procesoru.

Různé druhy programátorů pro programování mikroprocesorů:

- AVR Prog
- ISP – In System Programming
- STK500 – Zkratka programovacího protokolu
- The Dragon (USB)
- USBasp – USB programmer for Atmel AVR controllers

#### 5.3.1 Programátor AXIS PRESTO

Vybraný programátor je od firmy ASIX [12]. Tato firma, založená roku 1991, se zabývá se výrobou elektronických zařízení. Zaměření ASIX jsou programátory pro programovací studia, logické analyzátory nebo různé USB produkty.



Obrázek 9: Staré a nové provedení programátoru. [12]

Pro naprogramování procesoru byl použit programátor PRESTO. Je to nejnovější verze programátorů od ASIX a navazuje na předcházející verze, jako byly PICQUICK, PICCOLO nebo CAPR-PI.

Je to rychlý a flexibilní USB programátor (viz obr. 9/s. 29), který byl speciálně navržen pro programování integrovaných obvodů. Je zcela kompatibilní s AVR, ale i s dalšími typy, např. mikrokontroléry PIC, ARM a s aplikacemi Atmelu. Využívá se i k programování pamětí. Firma usiluje o to, aby programátor byl opravdu co nejrychlejší, takže je optimalizován k dosažení maximálních rychlostí při programování. Výrobce v dokumentaci [13] udává, že programátory PRESTO stihnou naprogramovat procesor do 3 s. Je napájen z USB a má velkou výhodu v tom, že obsahuje interní napájení pro programovatelnou součástku, takže není třeba řešit externí zdroj. Toto napájení se dá nastavit a také odpojit. Má vyvedených 8 pinů, z toho na pinu 3 VDD a pinu 1 VPP je nadproudová ochrana. Přepětová ochrana je pouze na VDD. Pin VPP je omezený 13 V.

K připojení programované součástky slouží 8pinový konektor ISP. VPP pracuje jako obousměrná datová komunikace a je připojen k napájecímu napětí. Toto napětí dosahuje buď 5 V z USB, nebo se jedná o napětí přiváděné z aplikace 3 až 5 V. Stejné napětí využívají i digitální signály. Rozšíření napětového rozsahu je možné pomocí nastavných hlavicek.

Na programátoru se nacházejí dvě LED diody. První zelená nás informuje o připojení



k USB a druhá žlutá o aktivitě programování. Tlačítko GO programování spouští.

Nejdříve je třeba nainstalovat ovladače programátoru do počítače. Ovladače jsou volně ke stažení na oficiálních stránkách firmy ASIX. Po instalaci ovladačů se instaluje software, který si uživatel vybere.

Základem je program UP, který podporuje celý rozsah výrobce ASIX a má mnoho možností použití programátoru. Jednou z nich je definování projektů, klávesové zkratky nebo bezobslužné použití. Program funguje na všech verzích Windows, ale i Linuxu. Funguje i podpora USB. Dalšími možnými programy jsou JTAG SVT Player, ARMINE nebo eCOG Programmer. Ty jsou vyrobeny pro speciální požadavky mikrokontrolérů ARM nebo eCOG. PRESTO poskytuje základní podporu při ladění aplikací a u některých mikrokontrolérů se jím dají řídit režimy součástky jako je Reset a Run. V některých případech i Stop.

### 5.3.2 Programovací jazyk C++

Jedná se o rozšíření objektově orientovaného jazyka C a podporuje několik programovacích stylů. V současnosti C++ spadá do nejrozšířenějších programovacích jazyků na světě.

Programovací jazyk C++ [11] byl vyvíjen od roku 1979 (jeho předek jazyk C několik let před ním) a za vynálezce tohoto jazyka je považován Bjarne Stroustrup. Jméno C++ vytvořil Rick Mascitti v roce 1983. Dvě znaménka za starším názvem C označují operátory inkrementace. Oficiální norma jazyku C++ byla uznána a přijata v roce 1998 a další aktualizace v následujících letech.

První preprocesory neboli překladače C++, překládaly jazyk do staršího C, aby se daly vůbec přechít. C++ není s C přímo kompatibilní, jak se může zdát. Můžeme narazit na program, který je naprogramován v jazyce C, ale v C++ ho nelze překládat. C++ se tedy používá pro programování aplikací a pro systémy plánování. Nejčastější a přímo typické aplikace, pro které je jazyk vyroben, jsou signálové procesory a virtuální stroje. Postupem času se objevují novější programovací jazyky, které C++ vytlačují, jako je například JAVA. C++ podporuje několik programovacích technik jako je strukturované programování, objektově-orientovaného programování nebo procedurální programování.



### 5.3.3 Programování

Programování procesoru probíhalo v AVR studiu v jazyce C++ [14].

Program začíná implementací knihoven, které jsou nutné pro příkazy a ty se dále využívají v programu. K nim je třeba deklarovat proměnné vstupy a výstupy pomocí příkazu `unsigned`. Na začátku programu probíhá výčet zadaných parametrů jako je dělička, registr a frekvence, které je třeba dobře nastavit pro správnou funkci emulátoru. V další části se nacházejí knihovny. U každé knihovny je napsáno, do jaké kategorie spadá a jakou v ní má úlohu. Na začátku je třeba vnutit procesoru jistou frekvenci, aby čítač nabýval rozumných hodnot. Zvolená frekvence má hodnotu 100 Hz.

První použitá procedura v programu čte hodnotu z A/D převodníku. V programu se načítá hodnota z převodníku ADC ze dvou různých kanálů, tedy přepínáním multiplexoru. Dále se inkrementuje přetečení časovače. V tomto případě od 0 do 59, s tím, že poslední dvě hodnoty (58 a 59) neovlivňují výstup. ADC načte hodnotu napětí z pedálu, tím se následně provede výpočet periody v závislosti na zatěžovateli, který může být proměnný. PWM zatěžovatel v dalším řádku se nastavuje od 0 do 100 %. V tomto případě je na 71 % pro správnou funkci generátoru. To se nastavuje v analogovém převodníku pod třetím kanálem. Program v přerušovacím cyklu rozeznává, zda jde o první či druhou půlperiodu. První půlperioda bude mít rozdílnou délku oproti druhé půlperiodě, což je závislé na poměru pulzů. Jedna bude 71 % a druhá zbylých 29 % v konkrétním případě. Je třeba konstantní perioda i při změně zatěžovatele PWM. Ta je závislá na sešlápnutí pedálu, takže musí být regulovatelná. Je použita i plná perioda, ta ovšem kontroluje součet obou půl period.

Další proměnnou jsou otáčky. Ty jsou v programu nastaveny pomocí vzorce na otáčky za minutu, jako má klasický otáčkoměr. Přepočet byl proveden proto, aby výsledek nebyl udáván v Hz, a bylo zřejmé na první pohled, o jaké hodnoty se jedná. V programu se dá libovolně měnit hodnota volnoběhu a otáček. To si uživatel může libovolně přizpůsobit, dle aplikace. Hodnoty jsou nastaveny na 800 ot/min volnoběh a 10000 maximální otáčky. Na takovou hodnotu maximálních otáček se v emulátoru nelze dostat. Dokumentace udává necelých 7000.

V hlavním programu se nejdříve provede nastavení vstupních a výstupních řadičů, jako jsou piny, časovač, nebo globální přerušování. Začíná se procedurou pro čtení převodníku, kde





procedura ADMUX určuje vybraný A/D kanál převodníku. V AVR studiu se nacházel pouze jeden analogově-digitální převodník, ale má možnost nastavení a přepínání kontaktů, takže je možno měřit na více kontaktech. ADMUX je tedy multiplexor, na kterém se čte analogová hodnota. V tomto případě připojená na pinu 2 a 3. Nastavují se tam parametry jako je rychlost čítání, zda je inverzní nebo neinverzní. Čekací smyčka zajistí čas A/D převodníku, aby stihl zapsat do paměti nastavenou hodnotu. Na konci smyčky je return ADCW a ten nám vypíše získanou hodnotu A/D převodníku, vyvoláním dané procedury. Jde o 10bitový ADC, takže dosahuje hodnot od 0 do 1023. Hodnotě 1023 odpovídá téměř napájecí napětí ( $U_{cc} - q$ , kde  $q = U_{cc}/1023$ ). Časovač je nastaven na CTC mód, který zajišťuje proměnnou frekvenci se střídou 1:1 na registru. Změna střídý (či zatěžovatele) je řešena střídáním hodnoty čítače v CTC módu, tato hodnota se vystřídá vždy při přetečení tohoto časovače, kde také detekujeme, o jakou periodu se jedná. Lze nastavit na mnoho dalších módů jako je Fast PWM, ale pro tuto aplikaci by bylo řešení náročnější na výpočetní výkon MCU.

Časovač pracuje při určité frekvenci a ta se dopočítává z přiložených vzorců v poznámce programu. Předem nastavených 10 MHz se dělí převodem, který je možno měnit. Tím můžeme zjistit ideální takt MCU. Kdyby byl převod 1 tak frekvence odpovídá otáčkám. Je tedy ideální hodnota 10 MHz. Při výpočtu program automaticky zaokrouhluje nahoru, takže mohou vznikat drobné časové rozdíly.

Převodník je nastaven na napětí, proti kterému má měřit. Může měřit proti referenčnímu nebo internímu. Vybrané je referenční, ale vše je libovolně nastavitelné.

Je třeba ještě správně nastavit poměr otáček proti frekvenci. Převod je nastaven na 100 kHz a volnoběh 600/převod. Tímto se zajistí, že 1 Hz odpovídá 100 ot/min. Maximální otáčky jsou vyřešeny úplně stejně, ale je třeba od nich odečíst volnoběh, kvůli algoritmu v programu. Je to proto, aby program nezačínal v nulové hodnotě, ale na nějakých minimálních otáčkách. Hodnota se nesmí dostat pod volnoběh a zároveň nesmí přesáhnout maximální otáčky. V této části probíhá opakující se for cyklus, který tvoří nekonečný cyklus výpočtů parametrů. Načtením kanálu 2 se načte uložená hodnota v pedálu a přirovná vstup hodnotě ADC. Přepočtená hodnota otáček je

$$(vstup/1023) * maxotacky + volnobeh.$$

To samé pro kanál 3, kterému je přiřazena PWM. To zastupuje zapojený odporový trimr,



jako dělič napětí. Vstup je také 0 – 1023, ale do PWM zatěžovatele je třeba přepočítat

$$(vstup/1023) * 100.$$

Tím vzniká možná regulace od 0–100 a nastaví se požadovaná hodnota. Pro tento případ 0–71. Vzorec je možno smazat a vložit přímou hodnotu 71.

V poslední části programu se na základě vstupních parametrů z A/D převodníku v každém přerušovacím cyklu přepočítává frekvence a střída. Vektor přerušení ISR odpočítává časy první půlperiody a druhé půlperiody. Až se obě naplní dle požadovaných parametrů, tak se přeruší. Otáčky se přepočítají z ot/min na ot/s, čímž můžeme počítat v Hz. První půlperioda se vypočte

$$(ot/60) * (PWMzatezovatel/100)/2$$

a druhá půlperioda

$$(ot/60) * (1 - (PWMzatezovatel/100))/2.$$

Rozdíl za jedničkou je zde, protože už v prvním kroku vyjde 0,71, takže automaticky to odečte a vychází 0,29. Obě periody se dělí dvěma, protože CTC udává frekvenci přetečení při zapnutí a vypnutí. V dalším kroku probíhá kontrola, že součet první půlperiody a druhé půlperiody dává plnou periodu.

Dále se inkrementuje proměnná pulz. První, druhý, třetí a dále, podle toho jak se přerušuje. Pokud je počet pulzů menší než 60, tak je další podmínka, která nastaví registr na 0–15 bitů. Tím se určí čas. Přepočet je

$$FCPU/(64 * prvnipulperioda).$$

Frekvence z procesoru je proměnná, takže se může libovolně nastavit a dělíme ji hodnotou 64 (pevně nastavená dělička) a první půlperiodou. Pokud jde o nultou půlperiodu, tak program pracuje úplně stejně, ale vloží tam druhou periodu.

V posledním řádku neguje stav pinu, a to pouze v případě, že je větší nebo roven 57. Jinak se s tímto pinem dále nepracuje, ale čítá pulzy dále. Pokud program dosáhne hodnoty 59, tak se vynuluje a opakuje rutinu.





## Závěr

Nejprve jsem se musel seznámit se samotným panelem elektroinstalace Škoda Superb 1,9 TDI modelu 2002. Množství kabelů a komponent na panelu bylo ze začátku poměrně složité na pochopení. Na většinu jsem postupem času přišel, nebo mi byly vysvětleny.

Moje práce spočívala v návrhu generátoru, který měl vytvořit signál odpovídající reálnému výstupnímu signálu ze snímače otáček. Na začátku práce bylo třeba pochopit samotnou funkci snímačů otáček ve voze. Jak pracují, podle čeho rozeznávají celkové otáčky a pochopit funkci snímačů.

První řešení generátoru spočívalo na časovači 555 a obsahovalo dva čítače typu CMOS. Toto řešení není zrovna krokem kupředu, nicméně jsem si chtěl ověřit funkčnost aplikací tohoto známého obvodu. Zapojení jsem navrhl v programu EAGLE a poté vyzkoušel jeho funkčnost v zapojení na nepájivém poli. Toto řešení ovšem nebylo funkční. Ačkoli časovač 555 pracoval podle osciloskopu správně, druhá část obvodu nikoli. Chybou byly špatně propojené vstupy čítačů a resetovací část. Důvody nefunkčnosti nebyly dále zkoumány, protože vedoucí práce doporučil řešení s použitím procesoru.

Přešel jsem tedy k druhému pokrokovějšímu řešení oproti předchozímu. Jednalo se o zapojení s programovatelným procesorem ATtiny45. Řešení jsem opět navrhl v prostředí EAGLE a tentokrát i zrealizoval jeho výrobu na desce plošných spojů, osadil navrženými součástkami a vytvořil program. Ten jsem poté programátorem nahrál do procesoru. Zde nastal problém s kompatibilitou samotného programátoru s PC. Po problémech se procesor podařilo konečně naprogramovat, ale následkem mnoha přepojování na desce plošných spojů byl SMD procesor z části poničen. A jelikož dodávka nového procesoru se zpozdila nebylo možné generátor jako celek otestovat a získat důkazy o funkčnosti navrženého zařízení.

V bakalářské práci jsem se seznámil s panelem elektroinstalace Škody Superb a navrhl dvě řešení generátoru signálu otáček motoru. Oboje řešení jsem zrealizoval a testoval. Bohužel se mi nepodařilo díky technickým problémům zprovoznit ani jedno z nich do termínu odevzdání této zprávy. Přesto si myslím, že druhé řešení je navrženo správně a bude jistě po nahrazení vadných součástek generovat signál podobný jako ze snímače otáček běžícího motoru.



## Reference

- [1] Skoda-auto.cz [online]. 2011 [cit. 2011-05-09]. Škoda Auto. Dostupné z WWW: <http://www.skoda-auto.cz/CZE/Pages/homepage.aspx>
- [2] Auto alarmy [online]. 2009 [cit. 2010-12-19]. Co je CAN BUS. Dostupné z WWW: <http://www.autoalarmy.cz/clanky/co-je-to-can-bus.html>
- [3] VÍT, JAKUB. *Emulátor spalovacího motoru vozu Škoda Superb*. TUL, 2010. 26 s. Ročníkový projekt. Technická univerzita Liberec.
- [4] Obvod 555 [online]. Náchod : Gymnázium Náchod, 2006 [cit. 2011-05-14]. Dostupné z WWW: <http://fyzika.gymnachod.cz/sem/texty/obvod555.doc>.
- [5] Agilent VISA. Agilent Technologies [online]. 1996, 5, [cit. 2011-05-14]. Dostupný z WWW: <http://cp.literature.agilent.com/litweb/pdf/E2090-90040.pdf>.
- [6] [Http://www.qrp.pops.net](http://www.qrp.pops.net) [online]. 2008 [cit. 2011-05-14]. Home Builder. Dostupné z WWW: <http://www.qrp.pops.net/LED-2008.asp>.
- [7] [Http://www2.atmel.com/](http://www2.atmel.com/) [online]. 2011 [cit. 2011-05-14]. Atmel. Dostupné z WWW: <http://www2.atmel.com/>.
- [8] Atmel.com [online]. 2011 [cit. 2011-05-09]. ATtiny 25/45/65. Dostupné z WWW: [http://www.atmel.com/dyn/resources/prod\\_documents/2586S.pdf](http://www.atmel.com/dyn/resources/prod_documents/2586S.pdf).
- [9] FERENC, Bohumil. [Www.Mjauto.cz](http://www.mjauto.cz) [online]. 2000 [cit. 2011-05-14]. Mjauto. Dostupné z WWW: [http://www.mjauto.cz/newdocs/ferenc/fer\\_mo02/fer\\_mo02.htm](http://www.mjauto.cz/newdocs/ferenc/fer_mo02/fer_mo02.htm).
- [10] Soubor:1-3-2007-Avr1.jpg. In Wikipedia : the free encyclopedia [online]. St. Petersburg (Florida) : Wikipedia Foundation, 1.3.2007 [cit. 2011-05-12]. Dostupné z WWW: <http://cs.wikipedia.org/wiki/Soubor:1-3-2007-Avr1.jpg>
- [11] C (programming language). In Wikipedia : the free encyclopedia [online]. St. Petersburg (Florida) : Wikipedia Foundation, 2010, last modified on 2011 [cit. 2011-05-14]. Dostupné z WWW: [http://en.wikipedia.org/wiki/C\\_\(programming\\_language\)](http://en.wikipedia.org/wiki/C_(programming_language)).



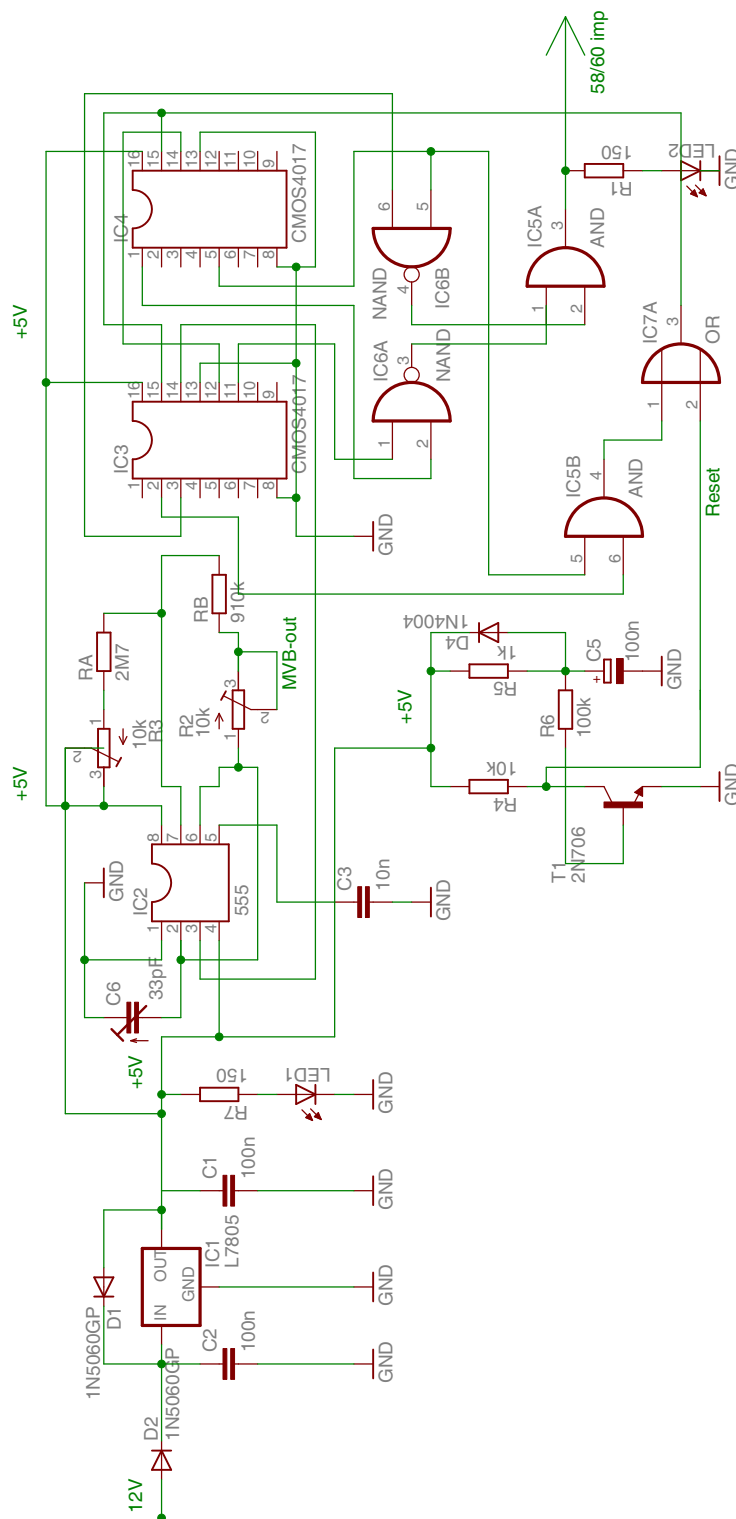
- [12] [Http://www.asix.cz/](http://www.asix.cz/) [online]. 2009 [cit. 2011-05-14]. ASIX. Dostupné z WWW: <http://www.asix.cz/>.
- [13] [Http://www.asix.cz/](http://www.asix.cz/) [online]. 1991 [cit. 2011-05-14]. Programátory ASIX. Dostupné z WWW: [http://www.asix.cz/download/programmers/programmers\\_cz.pdf](http://www.asix.cz/download/programmers/programmers_cz.pdf).
- [14] FOJTÍK, Rostislav. Programování v C [online]. TU : Univerzita Ostrava, 2004 [cit. 2011-05-17]. Dostupné z WWW: [http://www.umat.feec.vutbr.cz/~hlavicka/vyuka/Programovani\\_v\\_C\\_Fojtik.pdf](http://www.umat.feec.vutbr.cz/~hlavicka/vyuka/Programovani_v_C_Fojtik.pdf).

Poděkování: Tento materiál vznikl v rámci projektu ESF (CZ.1.07/2.2.00/07.0247)

**Reflexe požadavků průmyslu na výuku v oblasti automatického řízení a měření.**

Formát zpracování originálu: titulní list barevně, další listy včetně příloh barevně.

## A Návrh generátoru pomocí obvodu 555





## A.1 Seznam součástek (řešení s 555)

Tabulka 3: Seznam součástek (řešení s 555)

Označení	Popis součástky
C1, C2, C3	100nF keramický kondenzátor
C5	1 uF elektrolytický kondenzátor
C6	33 pF kapacitor
RA	2,7 M $\Omega$ rezistor metal. 0,4 W 1% RM0207
RB	910 k $\Omega$ rezistor metal. 0,4 W 1% RM0207
R1	150 $\Omega$ rezistor metal. 0,4 W 1% RM0207
R2, R3	10 k $\Omega$ odporový trimr, 10x10mm, horizontální
R4	10 k $\Omega$ rezistor metal. 0,4 W 1% RM0207
R5	1 k $\Omega$ rezistor metal. 0,4 W 1% RM0207
R6	100 k $\Omega$ rezistor metal. 0,4 W 1% RM0207
R7	150 R $\Omega$ rezistor metal. 0,4 W 1% RM0207
IC1	L7805 stabilizátor 5 V
IC2	Integrovaný obvod 555 časovač
IC3, IC4	CMOS 4017 čítač
IC5A, IC5B	4073 hradlo AND
IC6A, IC6B	4012 hradlo NAND
IC7A	4071 hradlo OR
T1	2N706 tranzistor TO92
D1, D2, D3	1N5060 usměrňovací dioda
LED1	Zelená led dioda 1,8–2,2 V 20 mA





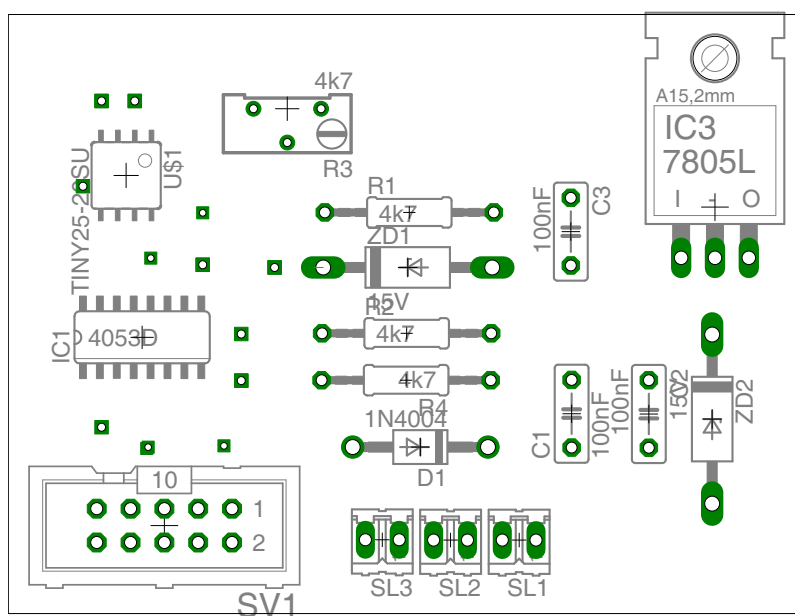
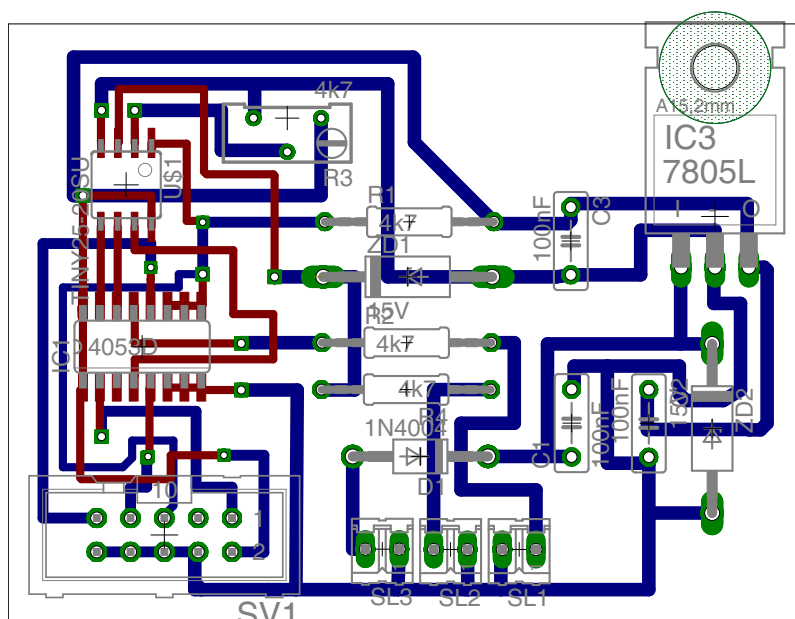
## B.1 Seznam součástek (řešení s ATtiny45)

Tabulka 4: Seznam součástek (řešení s ATtiny45)

Označení (ve schématu)	Popis součástky (hodnota, parametry)
C1, C2, C3	100 nF keramický kondenzátor
D1	1N4004 usměrňovací dioda
ZD1, ZD2	Zenerova dioda 15 V 0,5 W
IC1	4053D SO16
IC3	L7805 stabilizátor 5 V
R1, R2, R3, R4	4,7 k $\Omega$ Rezistor metal. 0,4 W 1% RM0207
SL1, SL2, SL3	Lišta kontakťová, 1řadá, přímá, pozlacená
US1	Procesor ATtiny45 SO08
R5	1 k $\Omega$ Rezistor metal. 0,4 W 1% RM0207
R6	100 k $\Omega$ Rezistor metal. 0,4 W 1% RM0207
R7	150 $\Omega$ Rezistor metal. 0,4 W 1% RM0207



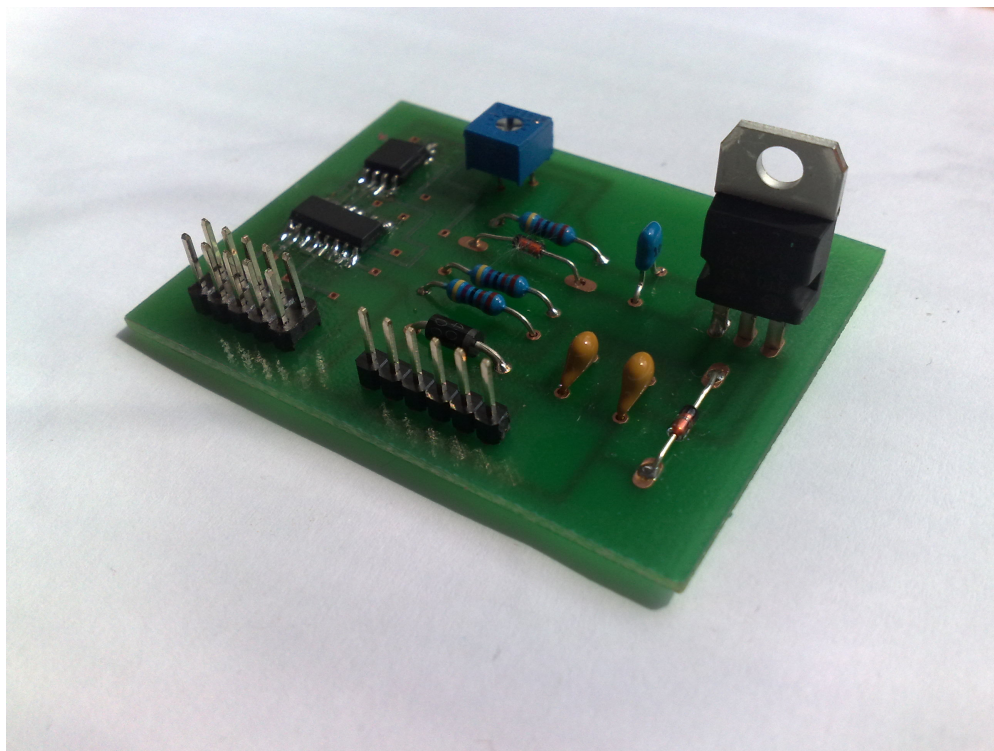
## B.2 Návrh plošného spoje pro generátor s procesorem ATtiny45







### B.3 Vyrobená deska generátoru s procesorem ATtiny45





## B.4 Textový výpis z programu AVR studio

```
#include <avr/io.h>           // knihovna pinu v~portech
#include <avr/interrupt.h>     // knihovna preruseni
#include <stdio.h>             // knihovna vstupu a~vystupu
#include <stdlib.h>            // knihovna prideleni pameti
#include <util/delay.h>        // knihovna zpozdeni

unsigned int vstup;           // hodnota z~ADC
unsigned int pulz = 0;        // pocet pulzu z~preruseni
unsigned int PWM_zatezovatel; // zatezovatel v % (0;100)
unsigned int pul_perioda;     // 0 či 1, urci o~kterou hranu jde

unsigned int otacky;          // otacky motoru za minutu (z~pedalu)
unsigned int volnobeh;        // volnobeh motoru za minutu
unsigned int max_otacky;      // maximální otacky motoru za minutu
                                // (nutno pricist volnobeh)
unsigned int prevod;          // pocet otacek na 1 Hz

unsigned int prvni_pul_perioda; // periody v~Hz!!!
unsigned int druha_pul_perioda;
unsigned int cela_perioda;

unsigned char kanal = 0;      // vyber analog. vstupu

unsigned int read_adc(unsigned char kanal) // analogovy vstup
{
    ADMUX = kanal;            // mereny kanal
    ADCSRA |= 0x40;           // parametry ADC
    while ((ADCSRA & 0x10)==0) // ceka nez precte hodnotu napeti
        // ADCSRA |= 0x10;
    return ADCW;
}

int main (void)
{
    DDRB |= (1 << 0);         // Nastavení výstupu
```



```

TCCR0B |= (1 << WGM00);          // Nastaví casovac 1 do CTC modu

TIMSK |= (1 << OCIE1A);          // Povolí CTC preruseni

sei();                            // Povolí globalni preruseni

// f = (F_CPU/(predelicka*OCR1A))
// OCR1A = (F_CPU/(predelicka*f))
// OCR1A = 15624; // Nastavi CTC komparator na 1Hz pri 1MHz AVR taktu,
// s~predelickou 64
// 10/prevod MHz, 64 dělička, 11718,75 citac -> 13,33 Hz -> 800 ot/min
// -> volnobeh
// 10/prevod MHz, 64 dělička, 937,5 citac -> 166,66 Hz -> 10000 ot/min
// -> otacky
// Pouze cela cisla do citace (OCR1A)!!! Nutno zaokrouhlit a~vznika % chyba

TCCR0B |= ((1 << CS10) | (1 << CS11)); // Pusti casovac na frekvenci Fcpu/64

ADMUX = ADC_VREF_TYPE;
ADCSRA = 0x87;

prevod = 100;          // 1Hz = 100ot/min
volnobeh = 600/prevod; // volnobeh - otacky za minutu
max_otacky = 7000/prevod - volnobeh; // maximalni otacky za minutu
// (max F_CPU/64/1000)

for (;;)              // samotny program
{
kanal = 2;            // Volba analog. z~pedalu
vstup = read_adc(kanal); // Nacte aktualni hodnotu napeti (0-1023)
otacky = (vstup/1023)*max_otacky+volnobeh;

kanal = 3;            // Volba analog z~trimru
vstup = read_adc(kanal); // Nacte aktualni hodnotu napeti (0-1023)
PWM_zatezovatel = (vstup/1023)*100; // Dle analogoveho vstupu se meni

```



```
                                // v~rozsahu (0;100)

    }

}

ISR(TIMER1_COMPA_vect)        // Provede se tato akce když nabude OCR1A
{
    pul_perioda = !pul_perioda;
    prvni_pul_perioda = (otacky/60)*(PWM_zatezovatel/100)/2;
    druha_pul_perioda = (otacky/60)*(1-(PWM_zatezovatel/100))/2;
    cela_perioda = prvni_pul_perioda + druha_pul_perioda;
    pulz = pulz + 1;           // Pocita pocet vystupnich impulzu
    if (pulz <= (60-1))
    {
        if (pul_perioda) OCR1A = (F_CPU/(64*prvni_pul_perioda));
        else OCR1A = OCR1A = (F_CPU/(64*druha_pul_perioda));
        if (!(pulz >= (58-1))) PORTB ^= (1 << 0); // Prepni vystup, krom 58 a~59
                                                // na PortB_1
    }
    else pulz = 0;
}
```



## C Obsah kompaktního disku (CD)

- Bakalářská práce v elektronické podobě.
- Návrh generátoru pomocí obvodu 555 v prostředí EAGLE verze 5.4.
- Návrh generátoru pomocí procesoru ATtiny45 v prostředí EAGLE verze 5.4.
- Navrhnutý program pro procesor v jazyce C v programu AVR studio.